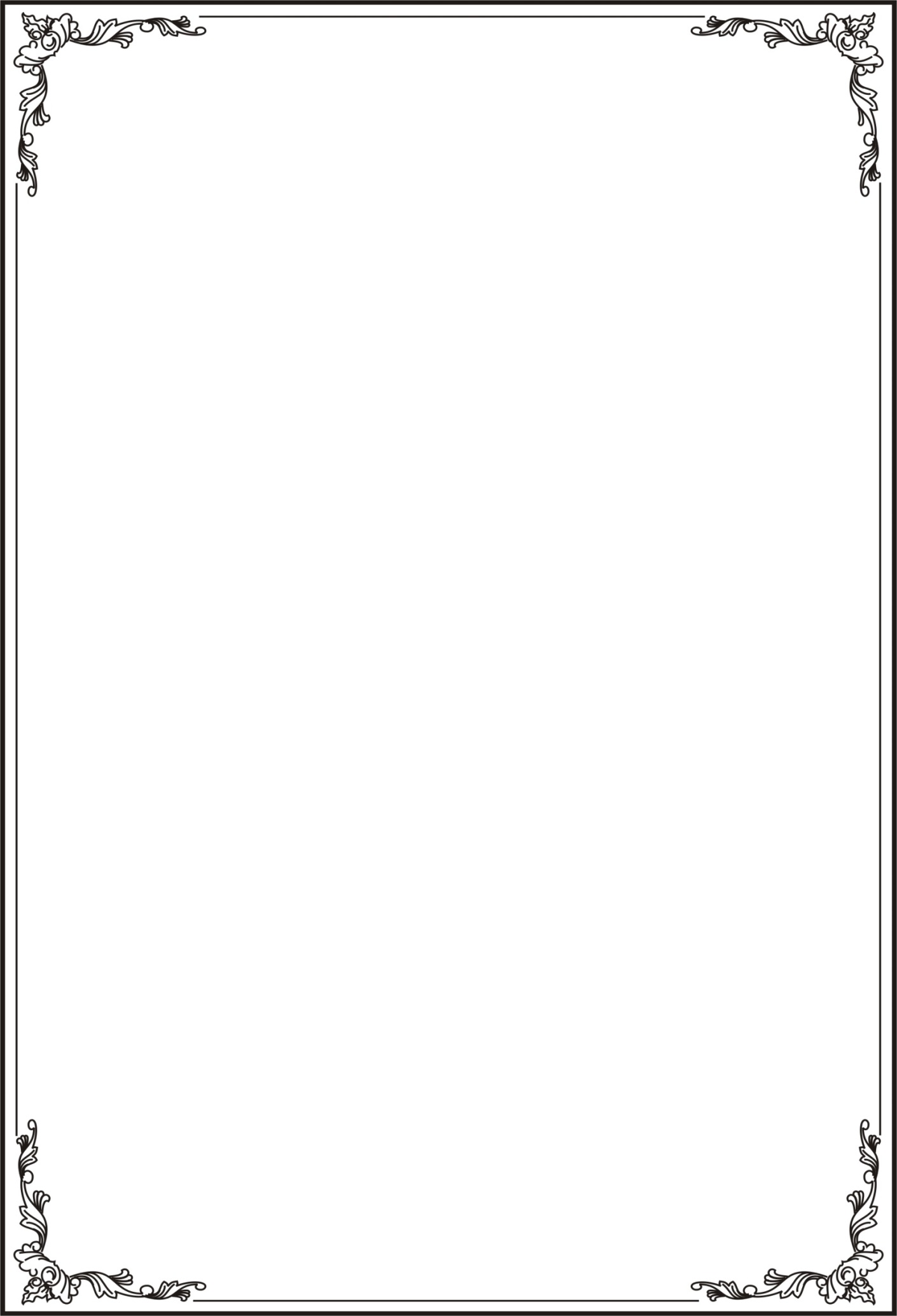
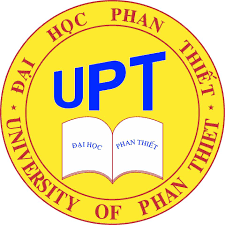
** BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC PHAN THIẾT**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------------**

****

**BÁO CÁO MÔN CÔNG NGHỆ PHẦN MỀM**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | ***ThS. Viên Thanh Nhã*** |
| **Sinh viên thực hiện:** | ***Lưu Quốc Nhân***  ***Ngô Minh Trường*** |
| **Mssv:** | ***K14THO0066***  ***K14THO0088*** |
| **Lớp:** | ***K14THO2*** |

**Phan Thiết, Tháng 6 năm 2024**

**MỤC LỤC**

[1.Khái niệm và mô hình thác nước(Waterfall Model) 1](#_Toc10215)

[1.1.Khái niệm công nghệ phần mềm 1](#_Toc3098)

[1.2.Hướng phát triển 1](#_Toc23383)

[1.3.Mô hìnhmô hình thác nước(Waterfall Model) 2](#_Toc29713)

[1.4.Điểm mạnh và Điểm yếu 3](#_Toc26823)

1.Khái niệm và mô hình thác nước(Waterfall Model)

# 1.1.Khái niệm công nghệ phần mềm

**Công nghệ phần mềm** ([tiếng Anh](https://vi.wikipedia.org/wiki/Ti%E1%BA%BFng_Anh" \o "Tiếng Anh): software engineering) là sự áp dụng một cách tiếp cận có hệ thống, có kỷ luật, và định lượng được cho việc phát triển, sử dụng và bảo trì [phần mềm](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m" \o "Phần mềm). Ngành học [kỹ sư phần mềm](https://vi.wikipedia.org/wiki/K%E1%BB%B9_s%C6%B0_ph%E1%BA%A7n_m%E1%BB%81m" \o "Kỹ sư phần mềm) bao trùm kiến thức, các công cụ, và các phương pháp cho việc định nghĩa [yêu cầu](https://vi.wikipedia.org/wiki/Y%C3%AAu_c%E1%BA%A7u_(k%E1%BB%B9_thu%E1%BA%ADt)" \o "Yêu cầu (kỹ thuật)) phần mềm, và thực hiện các tác vụ [thiết kế](https://vi.wikipedia.org/wiki/Thi%E1%BA%BFt_k%E1%BA%BF_ph%E1%BA%A7n_m%E1%BB%81m" \o "Thiết kế phần mềm), [xây dựng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh" \o "Lập trình máy tính), [kiểm thử](https://vi.wikipedia.org/wiki/Ki%E1%BB%83m_th%E1%BB%AD_ph%E1%BA%A7n_m%E1%BB%81m" \o "Kiểm thử phần mềm) (software testing), và [bảo trì phần mềm](https://vi.wikipedia.org/wiki/B%E1%BA%A3o_tr%C3%AC_ph%E1%BA%A7n_m%E1%BB%81m" \o "Bảo trì phần mềm).Kỹ sư phần mềm còn sử dụng kiến thức của các lĩnh vực như [kỹ thuật máy tính](https://vi.wikipedia.org/wiki/K%E1%BB%B9_thu%E1%BA%ADt_m%C3%A1y_t%C3%ADnh" \o "Kỹ thuật máy tính), [khoa học máy tính](https://vi.wikipedia.org/wiki/Khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh" \o "Khoa học máy tính), [quản lý](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD" \o "Quản lý), [toán học](https://vi.wikipedia.org/wiki/To%C3%A1n_h%E1%BB%8Dc" \o "Toán học), [quản lý dự án](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD_d%E1%BB%B1_%C3%A1n" \o "Quản lý dự án), [quản lý chất lượng](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%BD_ch%E1%BA%A5t_l%C6%B0%E1%BB%A3ng" \o "Quản lý chất lượng), [công thái học](https://vi.wikipedia.org/wiki/C%C3%B4ng_th%C3%A1i_h%E1%BB%8Dc" \o "Công thái học) phần mềm (software ergonomics), và [kỹ sư hệ thống](https://vi.wikipedia.org/w/index.php?title=K%E1%BB%B9_s%C6%B0_h%E1%BB%87_th%E1%BB%91ng&action=edit&redlink=1" \o "Kỹ sư hệ thống (trang không tồn tại)) (systems engineering).

# 1.2.Hướng phát triển

Việc phát triển công nghệ phần mềm hiện nay đang tiếp tục tiến xa với những xu hướng và thách thức mới. Dưới đây là một số hướng phát triển quan trọng mà công nghệ phần mềm có thể đi theo trong thời gian tới:

1. **Trí tuệ nhân tạo (AI) và Machine Learning (ML):** AI và ML đang trở thành phần không thể thiếu trong nhiều ứng dụng phần mềm. Việc sử dụng các thuật toán này giúp cải thiện tự động hóa, dự đoán, và tối ưu hóa các quy trình.
2. **Edge Computing:** Phát triển ứng dụng tích hợp trí tuệ nhân tạo và tính toán trên các thiết bị đầu cuối như điện thoại di động và cảm biến, giúp giảm độ trễ và tăng cường bảo mật.
3. **Internet of Things (IoT):** Sự phát triển của IoT tạo ra một hệ sinh thái kết nối các thiết bị thông minh, yêu cầu việc phát triển phần mềm để quản lý, thu thập dữ liệu và tương tác với các thiết bị này.
4. **Blockchain:** Blockchain không chỉ được áp dụng trong tiền điện tử mà còn trong nhiều lĩnh vực khác như bảo mật, quản lý chuỗi cung ứng, và hợp đồng thông minh. Công nghệ này đòi hỏi phát triển phần mềm phức tạp để triển khai và quản lý.
5. **Phát triển phần mềm dựa trên nền tảng đám mây (Cloud Computing):** Đám mây đang trở thành một nền tảng phổ biến cho việc triển khai ứng dụng. Phát triển các ứng dụng dựa trên đám mây đòi hỏi hiểu biết sâu rộng về các dịch vụ như AWS, Azure, hay Google Cloud.
6. **Phát triển ứng dụng di động:** Sự bùng nổ của các thiết bị di động đã tạo ra một nhu cầu lớn cho các ứng dụng di động. Việc phát triển phần mềm cho các nền tảng di động như iOS và Android vẫn là một lĩnh vực có nhiều cơ hội.
7. **Phát triển phần mềm dựa trên mô hình DevOps:** DevOps không chỉ là một phương pháp phát triển phần mềm mà còn là một triết lý quản lý vận hành hệ thống. Phát triển phần mềm hiện nay cần tích hợp liên tục và triển khai liên tục (CI/CD) để cung cấp các ứng dụng và dịch vụ một cách nhanh chóng và linh hoạt.
8. **Bảo mật phần mềm:** Với sự gia tăng về tần suất và tình chất của các cuộc tấn công mạng, bảo mật phần mềm trở thành một yếu tố không thể phớt lờ. Phát triển các phương pháp và công nghệ bảo mật mới để bảo vệ ứng dụng và dữ liệu người dùng là một ưu tiên hàng đầu.

# 1.3.Mô hìnhmô hình thác nước(Waterfall Model)

Mô hình thác nước là một ví dụ của mô hình Sequential (Tuần tự). Trong mô hình này, hoạt động phát triển phần mềm được chia thành các giai đoạn khác nhau và từng giai đoạn bao gồm hàng loạt các nhiệm vụ và có các mục tiêu khác nhau.

Mô hình Thác nước là giai đoạn đầu trong quá trình SDLC. Trên thực tế, nó là mô hình đầu tiên được sử dụng rộng rãi trong ngành công nghiệp phần mềm. Nó được chia thành các pha, đầu ra của một pha trở thành đầu vào của pha tiếp theo. Nó là giai đoạn bắt buộc được hoàn thành trước khi bắt đầu giai đoạn tiếp theo. Nói tóm lại, không có sự chồng chéo nào trong mô hình thác nước.

Trong thác nước, sự phát triển của một pha chỉ bắt đầu khi giai đoạn trước hoàn thành. Do tính chất này, mỗi giai đoạn của mô hình thác nước phải được xác định khá chính xác. Các giai đoạn chuyển từ mức cao xuống mức thấp hơn, giống như một thác nước nên mô hình này được đặt tên là mô hình thác nước.

# 1.4.Điểm mạnh và Điểm yếu

* Điểm mạnh
* Dễ hiểu và sử dụng: Mô hình thác nước có cấu trúc rõ ràng và tuần tự, dễ dàng hiểu và triển khai. Điều này giúp các thành viên trong dự án dễ dàng theo dõi tiến độ và hiểu được mục tiêu của từng giai đoạn.
* Quản lý dự án dễ dàng: Vì mỗi giai đoạn được thực hiện một cách tuần tự và rõ ràng, việc quản lý dự án trở nên đơn giản hơn. Công việc có thể được phân chia và giao cho các nhóm chuyên môn một cách dễ dàng.
* Kiểm soát tiến độ: Với mỗi giai đoạn được hoàn thành trước khi bắt đầu giai đoạn tiếp theo, việc kiểm soát tiến độ dự án trở nên hiệu quả. Điều này giúp giảm thiểu rủi ro và xác định được vấn đề sớm.
* Điểm yếu
* Khó thích ứng với sự thay đổi: Mô hình thác nước không linh hoạt đối với sự thay đổi yêu cầu từ khách hàng. Một khi một giai đoạn đã hoàn thành và chuyển sang giai đoạn tiếp theo, việc thay đổi yêu cầu có thể gây ra những vấn đề lớn và tăng chi phí.
* Khó khăn trong phản hồi sớm: Vì mỗi giai đoạn phải hoàn thành trước khi bắt đầu giai đoạn tiếp theo, việc phản hồi từ khách hàng hoặc người dùng có thể bị trì hoãn, khiến cho việc sửa lỗi và thay đổi trở nên phức tạp và tốn kém hơn.
* Rủi ro cao: Nếu có một lỗi được phát hiện ở giai đoạn sau này của quy trình, việc sửa chữa có thể trở nên đắt đỏ và tốn thời gian. Điều này có thể dẫn đến việc vượt quá ngân sách và thời gian dự án.